Data in Brief 48 (2023) 109068

Contents lists available at ScienceDirect

Data in Brief

journal homepage: www.elsevier.com/locate/dib

Data Article

Dataset for identifying maintenance needs of home appliances using artificial intelligence

Tiago Fonseca[a,*], Pedro Chaves[a], Luis Lino Ferreira[a], Nuno Gouveia[b], David Costa[c], André Oliveira[d], Jorge Landeck[e]

[a] School of Engineering of the Polytechnical Institute of Porto, Rua Dr. António Bernardino de Almeida, 431, 4249-015 Porto Porto, Portugal
[b] Sonae MCretail, SGPS, S.A., Rua João Mendonça, 529, 4464-501 Senhora da Hora, Matosinhos, Portugal
[c] WORTEN - EQUIPAMENTOS PARA O LAR, S.A., Rua João Mendonça n.° 505, Senhora da Hora, Portugal
[d] Cleanwatts, Ladeira da Paula 6, 3040-574 Coimbra, Portugal
[e] LIBPhys, Department of Physics, University of Coimbra, P-3004 516 Coimbra, Portugal

ARTICLE INFO

Article history:
Received 12 February 2023
Revised 8 March 2023
Accepted 10 March 2023
Available online 17 March 2023

Dataset link: SMART-PDM Appliance Dataset (Original data)

Keywords:
Data
Predictive maintenance
Appliances
Machine Learning

ABSTRACT

The ability to predict the maintenance needs of machines is generating increasing interest in a wide range of industries as it contributes to diminishing machine downtime and costs while increasing efficiency when compared to traditional maintenance approaches. Predictive maintenance (PdM) methods, based on state-of-the-art Internet of Things (IoT) systems and Artificial Intelligence (AI) techniques, are heavily dependent on data to create analytical models capable of identifying certain patterns which can represent a malfunction or deterioration in the monitored machines. Therefore, a realistic and representative dataset is paramount for creating, training, and validating PdM techniques. This paper introduces a new dataset, which integrates real-world data from home appliances, such as refrigerators and washing machines, suitable for the development and testing of PdM algorithms. The data was collected on various home appliances at a repair center and included readings of electrical current and vibration at low (1 Hz) and high (2048 Hz) sampling frequencies. The dataset samples are filtered and tagged with both normal and malfunction types. An extracted features dataset, corresponding to the collected working cycles is also made available. This dataset could bene-

* Corresponding author.
E-mail address: calof@isep.ipp.pt (T. Fonseca).

https://doi.org/10.1016/j.dib.2023.109068

fit research and development of AI systems for home appliances' predictive maintenance tasks and outlier detection analysis. The dataset can also be repurposed for smart-grid or smart-home applications, predicting the consumption patterns of such home appliances.

Specifications Table

Subject	Artificial Intelligence
Specific subject area	Predictive Maintenance of home appliances with machine learning and deep learning techniques
Type of data	Table
How the data were acquired	Raw data was collected from washing machines and fridges in a repair center located in Lisbon, Portugal by a specially developed MQTT sensor connected to these appliances. The sensor acquires timestamped power and vibration readings, at both low (1 Hz) and high sampling frequencies (2048 Hz). As generally most of the appliances' timespan is spent in standby, the middleware only activates and stores high sampling frequency readings when triggered by the analysis of the slow data (i.e., fast stream is activated as soon as an appliance working cycle is started). The filtered dataset, as well as the extracted features, were obtained through a special big data system introduced at [1].
Data format	Raw Analyzed Filtered
Description of data collection	The dataset contains 110 cycles of washing machines and 1111 cycles of refrigerators at a low sampling frequency (1 Hz). This dataset contains the active power and apparent power, among others. For the same cycles it also contains at an high sampling frequency (2048 Hz), the readings of vibration and current. The working cycles collected at the repair center encompass both malfunction situations and normal situations. The data was collected between the end of 2021 and the beginning of 2022, processed, and filtered. Files with the extracted features are also included.
Data source location	Institution: School of Engineering of the Polytechnical Institute of Porto City/Region: Porto, Portugal Country: Portugal
Data accessibility	Repository name: Zenodo Data identification number: DOI: 10.5281/zenodo.7245198 Direct URL to data: https://doi.org/10.5281/zenodo.7245198 [2]
Related research article	Peer Reviewed related research article: P. Chaves et al., "An IoT Cloud and Big Data Architecture for the Maintenance of Home Appliances," IECON 2022 – 48th Annual Conference of the IEEE Industrial Electronics Society, 2022, pp. 1-6, doi:10.1109/IECON49645.2022.9968580 [1]

Value of the Data

- The dataset is useful as it provides pre-filtered and annotated real-world data of both malfunctioning and normal working home appliances, such as washing machines and fridges, taken at a home appliances repair center.
- The dataset could benefit the work of researchers, data scientists, and companies studying predictive maintenance techniques for home appliances.

- The dataset contains a large set of extracted features for each cycle, making it very useful for the easy application of supervised and unsupervised classification and regression models by machine learning and deep learning algorithms, predicting the types of failures or when they are likely to happen. The algorithms can be applied to remotely identify component failures, proactively streamlining the technicians' repairs.
- Researchers can reuse the dataset, adding it to other collected data in the area, and ultimately creating a more complete and extensive dataset on home appliances data. The data can also be used for researchers to validate or compare the results against their own.
- Besides predictive maintenance, home appliance power consumption readings can be repurposed for smart grid or smart-home applications. For example, the dataset can be applied to train algorithms that predict the consumption patterns of such home appliances, which can be applied to energy flexibility scheduling, such as the work developed in [3].

1. Objective

The generation of this dataset came from the necessity of researchers inserted at the European project Smart-PDM [4] to detect, analyze and predict maintenance necessities of everyday use home appliances in domestic households. The aim of the project was not only to cut the downtime of the appliances by early detecting malfunctions but also to reduce maintenance difficulties and costs by helping technicians in identifying the specific malfunction. Due to the scarcity of detailed data in this area, the team set out to build, program, and test a big data collection and filtering infrastructure presented in the peer reviewed article [1], which enabled the collection of this dataset. This data article adds value to the mentioned publication as it proves its usability and shows the results of the use of such a platform in real-world scenarios. This paper also makes data collected available for further exploration, providing research opportunities and future work in this area.

2. Data Description

The dataset presented in this paper comprises raw and filtered data collected from washing machines and fridges at an appliance repair center between the end of 2021 and the beginning of 2022. Data was collected using a sensor capable of measuring active power, reactive power, apparent power, frequency, power factor, voltage, and current at 1 Hz (slow streams) and vibration and current at 2048 Hz (fast streams). The data was pre-filtered, meaning that from all the collected raw data, only cycles with a known status transmitted by the repair center technicians, are present in the dataset. As such, the dataset encompasses annotated normal and malfunctioning working cycles. The dataset is separated into three folders: (1) a folder for code scripts, (2) a folder for washing machine data samples, and (3) a folder for fridge samples.

2.1. Code Folder

The folder /code contains code scripts used to work with the data. Inside this folder, there are three files used during the dataset creation and analysis. filter_cycles_script.py - Script used for identifying the beginning and ending times of home appliances' working cycles. It searches the raw slow stream Active Power measurements and finds whether the appliance is running or not. In the end, this script returns a list with the begin and end times of each cycle. exctract_features.py - Script used for extracting features from time series data measured from home appliances. pca.py - Script used for doing a Principal Component Analysis (PCA) of the features of the datasets. A PCA is a statistical technique for analyzing datasets containing large numbers of features per observation. It increases the interpretability of data by enabling its multidimensional visualization while preserving the maximum amount of information [5].

Table 1
Data structure at the slow.csv files.

Column Names	Description
timestamp	When the data was measured (UNIX Timestamp)
ActE	The measurement of the active energy counter
ActP	The measurement of the active power
RctP	The measurement of the reactive power
AppP	The measurement of the apparent power
Fr	The measurement of the frequency
PF	The measurement of the power factor
V	The measurement of the voltage
A	The measurement of the current

Table 2
Data structure at the fast.csv files.

Column Names	Description
timestamp	When the data was measured (UNIX Timestamp).
Current	The measurement of the current power
Vibration	The measurement of the vibration

2.1.1. Washing Machines Folder

Inside the /washing_machines folder there are the filtered data, metadata, and extracted features of the samples collected from real-world washing machines. The folder structure is as follows:

```
/beginX_endX
  slow.csv
  fast.csv
/beginY_endY
  slow.csv
  fast.csv
...
washing_machine_metadata.csv
WM_ExtractedFeatures.csv
```

First, each cycle data is divided into a subfolder. Each subfolder is named according to the following nomenclature: begin_end. The *begin* and *end* are the unique identifiers of the cycle and match the washing machine cycle start and end dates and time in a human-readable format. Inside these subfolders, there are two Comma-Separated Values (CSV) files. The slow stream (files with the name slow) are measurements at one sample per second of the active power, reactive power, apparent power, frequency, power factor, voltage, and current of the appliance. Data inside slow.csv files follow the structure presented in Table 1.

The fast streams (files with the name fast) are sampled at 2048 Hz, but now only for Current and Vibration. Fast.csv files follow the structure presented in Table 2.

Second, the washing_machine_metadata.csv file labels each washing machine cycle. The file format follows the following structure.

Third, the WM_ExtractedFeatures.csv file contains extracted features for each working cycle in the folder. The process and code used to extract features are in the /code folder previously introduced. With such code, more and different features can be extracted. Note that this file contains features extracted for both slow Active Power and fast current and vibration stream data.

Table 3
Metadata file structure.

Column Names	Description
begin_end	Washing machine working cycle start and end date and time in a human-readable format. Matches the name of the corresponding cycle subfolder
timestamp_begin	When the washing machine working cycle begins (UNIX Timestamp).
timestamp_end	When the washing machine working cycle ends (UNIX Timestamp).
brand	The brand of the machine.
model	The model of the machine
program	The washing cycle program
temperature	The temperature used during the cycle
spin	The chosen centrifuge spin of the washing cycle
load	The weight of the clothes loaded into the machine
failure	If the machine is working properly or if it had any known failure. Also known as the target variable. This variable can take the values: • Normal • Heating • Bearings • Motor and supports

2.1.2. Refrigerators Folder

Similar to the washing machines folder, the refrigerators folder contains the filtered data, metadata, and extracted features of real-world refrigerators. The folder structure is as follows:

```
/beginX_endX
  slow.csv
  fast.csv
  24h.csv
/beginY_endY
  slow.csv
  fast.csv
  24h.csv
...
fridge_metadata.csv
F_ExtractedFeatures.csv
```

Once again, each subfolder corresponds to a cycle and is named according to the nomenclature: begin_end. The *begin* and *end* are the unique identifiers of the cycle and match the fridge cycle start and end dates and daytime in a human-readable format. Inside these subfolders, there are three CSV files. As in the washing machines, these are the slow streams (slow.csv) and the fast streams (fast.csv) which follow the structure in Tables 1 and 2. However, refrigerators have a third file, the 24h.csv. This file contains the slow stream of the last 24 hours of the refrigerator. Our team decided to include this file for the refrigerators and not for washing machines, due to the continuous operation of refrigerators across the day, which differs from the more sporadic utilization nature of washing machines. Data inside 24h.csv files follow the same structure as the slow.csv files, presented in Table 1.

The fridge_metadata.csv file labels each cycle inside the folder. The file format follows the following structure for refrigerators.

Similar to washing machines (Table 4), the ExtractedFeatures.csv file contains, at each line, the extracted features for each recorded cycle, as shown previously in Table 4. However, in fridges, the same array of features was also extracted for the 24h.csv files and added to each cycle array of features (each line of the F_ExtractedFeatures.csv file).

Table 4
ExtractedFeatures file structure and description.

Data From which features are extracted	Feature name	Feature description
Slow Stream Active Power (Prefix ActP), Fast Stream Current (Prefix Current), Fast Stream Vibration (Prefix Vibration)	variance_larger_than_standard_deviation	A boolean value is equal to 1 if the variance of the time series is greater than its standard deviation. A value equal to 0 otherwise
	sum_values	Sum of all the time series values
	mean_abs_change	Average of the absolute differences between subsequent time series values
	mean_change	Average of subsequent time series value differences
	Median	The median of the time series
	Mean	The mean of the time series
	Length	The length of the time series
	standard_deviation	The standard deviation of the time series
	variation_coefficient	The ratio of the standard deviation to the mean of the time series
	Variance	The variance of the time series
	Skewness	The skewness of the time series (calculated with the adjusted Fisher-Pearson standardized moment coefficient G1).
	Kurtosis	The kurtosis of the time series (calculated with the adjusted Fisher-Pearson standardized moment coefficient G2).
	root_mean_square	The root mean square of the time series.
	count_above_mean	The number of values in the time series that are higher than the mean of the time series
	count_below_mean	The number of values in the time series that is lower than the mean of the time series
	maximum	The highest value of the time series
	absolute_maximum	The highest absolute value of the time series
	minimum	The first location of the minimal value of the time series
	number_peaks__n_5	The number of peaks of at least support 5 in the time series
	fft_aggregated__aggtype_""centroid"""	The spectral centroid (mean), variance, skew, and kurtosis of the absolute Fourier transform spectrum
	fft_aggregated__aggtype_""variance"""	
	fft_aggregated__aggtype_""skew"""	
	fft_aggregated__aggtype_""kurtosis"""	
	fourier_entropy__bins_5	The binned entropy of the power spectral density of the time series (using the welch method).
	fourier_entropy__bins_10	

3. Experimental Design, Materials, and Methods

3.1. Data Collection, Filtering, and Feature Extraction Overview

The data collection system was prepared to gather data from home appliances in households using a new sensor developed by a consortium partner, which communicated over an MQTT interface [1]. The system collects timestamped active power, reactive power, apparent power, frequency, power factor, voltage, and current measurements at slow sampling frequencies (1 Hz), and current and vibration at high sampling frequencies (2048 Hz). The dataset was then built by

Table 5
Refrigerator metadata file structure.

Column Names	Description
begin_end	Refrigerator working cycle start and end date and time in a human-readable format. Matches the name of the corresponding cycle subfolder
timestamp_begin	When the washing machine working cycle begins (UNIX Timestamp).
timestamp_end	When the washing machine working cycle ends (UNIX Timestamp).
brand	The brand of the machine.
model	The model of the machine
failure	If the machine is working properly or if it had any known failure. Also known as the target variable. This variable can take the values: • Normal • Malfunction

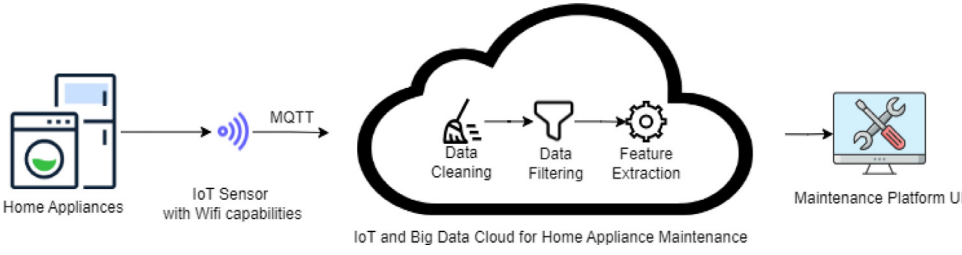

Fig. 1. Data collection phases overview.

channeling the data through three key steps in the data processing: (1) Data Cleaning, (2) Data Filtering, and (3) Feature Extraction. Fig. 1 shows an overview of the data collection system and its respective modules.

After collection, the data was cleaned in the Data Cleaning module. In this module, filters to remove noise and outliers were applied. Other operations included sorting the data according to its timestamp and removing duplicate readings. This module is also used to detect holes in the data, which are usually due to a temporary lack of connection with the sensors.

Second, at the data filtering step, as generally most of the appliances' timespan is spent on standby, the data is filtered, selecting only the time corresponding to working cycles. Then this information was matched with the archives from the repair center. This guarantees that the cycles in the dataset are always tagged with normal or specified malfunctions, which is crucial for the application of machine learning techniques. Filtered data at the subfolders of both washing machines and fridges, is under these circumstances, as seen before. The Python script that identifies cycles in the slow stream data can be found in the /code folder, under the name *filter_cycles_script.py*.

Third, to ease the use of the dataset in machine learning and deep learning applications, both a ready-to-use features file and the code to generate them are provided. As seen before, extracted features can range from very simple ones, such as the maximum, average, and minimum current and vibration to more complex ones, like Fast Fourier Transforms (FFT), Skewness, and Kurtosis.

The Python script that extracts features from the dataset data files can be found in the /code folder, under the name *extract_features.py*. This code is based on the feature extraction framework presented in [6].

Next, the features can be concatenated, grouping the features extracted for the current slow stream with the ones extracted for the current and vibration fast streams into a single feature array that translates the appliance cycle, the *ExtractedFeatures.csv* file for both washing machines and fridges. In the case of the fridges, and as seen before, also the features extracted from the last 24h of current data are aggregated in the final features.csv file. Note that on this dataset no

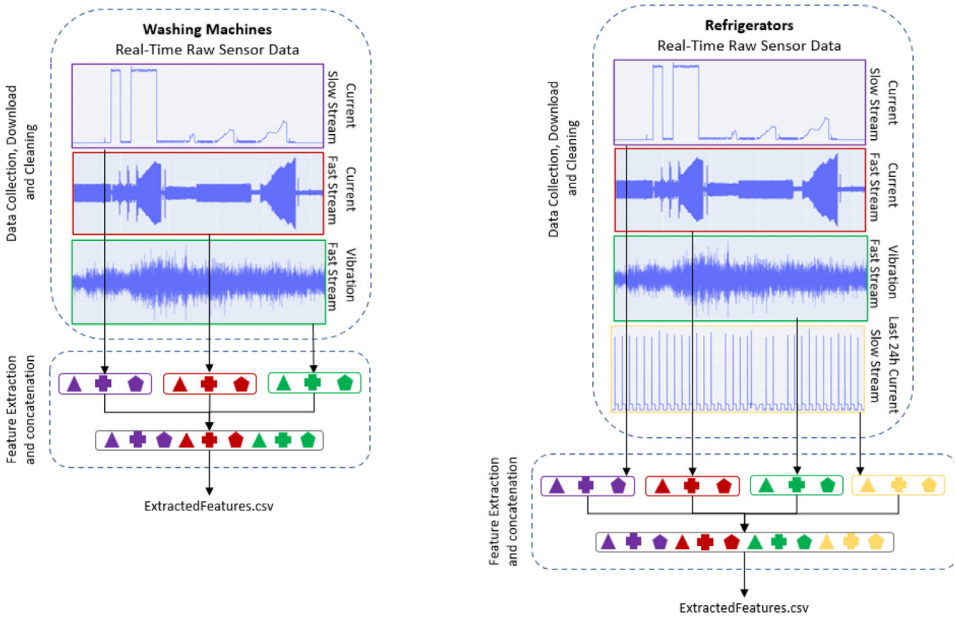

Fig. 2. Feature Extraction for washing machines (left) and refrigerators (right). Modified from [1].

feature selection phase was executed, allowing for a broader range of its study and utilization on other works. Fig. 2 overviews the feature extraction process applied to create this dataset.

Finally, features can be used to train machine learning classification models that classify if the device needs maintenance and what type of failure is happening. Outlier detection or time series forecasting can also be applied to the data.

3.2. Classification

The presented dataset can be used for a variety of Machine Learning and Deep Learning classification tasks. From foreseeing if a washing machine or fridge is working correctly or not, in a binary classification problem, predicting what specific type of failure will occur, what brand, or what type of washing program is occurring on a given washing machine, in a multiclass classification problem.

For example, a predictive model capable of correctly identifying what type of failure a washing machine is experiencing, at the consumer dwelling, allows for the technicians to be better prepared, taking with them replacing parts when going to carry out the repair. This strategy can optimize maintenance and reduce costs, as technicians do not have to take the first visit to diagnose the problem and further trips to repair the appliance.

Moreover, by developing models capable of estimating the occurrence of a failure at home appliances before it arises, certain predictive maintenance plans can be put in place. This enables, for example, the diminishing of further damages, keeping energy efficiency levels, and ultimately the reduction of costs for all parties involved.

In [1], machine learning algorithms were trained with the washing machine data collected for this dataset. Some machines were under normal operation conditions, while others have malfunctions, such as heating and bearing problems. Fig. 3 shows the feature importance, using a random forest algorithm, for this classification problem.

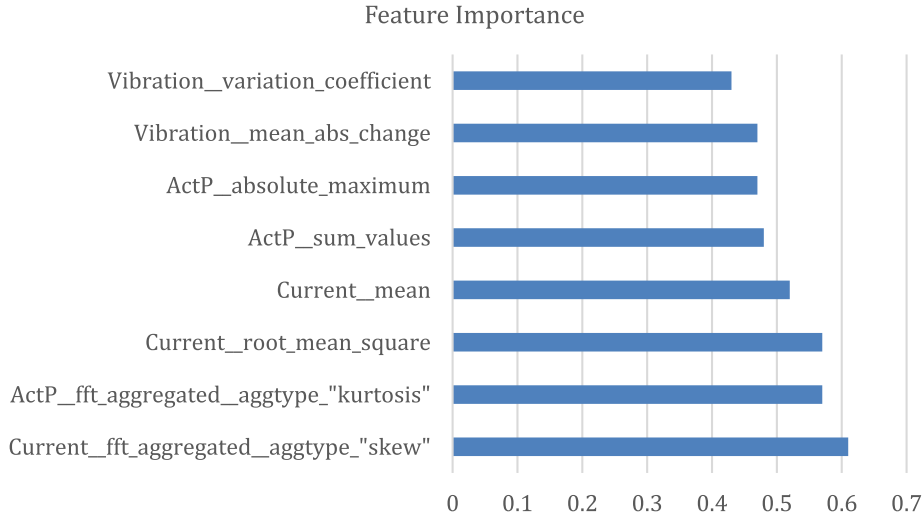

Fig. 3. The most important features for the detection of home appliances malfunctions, obtained from the use of a random forest classifier in the introduced dataset.

3.3. Outlier Detection

Outlier Detection is a technique used to identify occurrences that are outside of normal and expected behaviors. With applications in cyber-security intrusion detection and fraud detection, outlier detection can also play an important role in PdM systems. Most of the time collecting and identifying data from malfunctioning equipment is hard, given the sporadic nature and infrequent occurrence of such events. This technique serves, when applied to maintenance, as a first analysis to denote something needing attention, or out of normal operation, which can

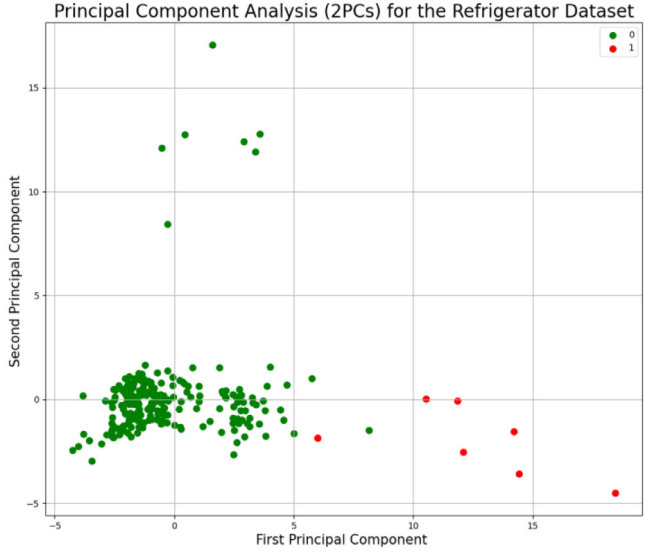

Fig. 4. 2-Dimensional PCA of fridges features dataset.

later be classified and fed for training in a classification model. The data collected for refrigerators in the present dataset, is unbalanced, with only a little percentage of samples being from the malfunction class. As such, this data suits the application of outlier detection techniques. For visualization, a PCA was applied, reducing the multi-dimensional data of the features dataset into 2-dimensional data which can be plotted in a 2D plot. Fig. 4 shows this plot, where green represents cycles of refrigerators under normal operation, and red the malfunction cycles. The Python script that performs a PCA analysis of the features dataset can be found in the /code folder, under the name *pca.py*.

Ethics Statements

This work did not involve any human or animal subjects, nor data from social media platforms.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Data Availability

SMART-PDM Appliance Dataset (Original data) (Zenodo).

CRediT Author Statement

Tiago Fonseca: Conceptualization, Software, Validation, Formal analysis, Investigation, Data curation; **Pedro Chaves:** Writing – original draft, Visualization, Visualization; **Luis Lino Ferreira:** Conceptualization, Validation, Investigation, Writing – review & editing, Supervision, Funding acquisition, Project administration; **Nuno Gouveia:** Conceptualization, Supervision, Resources, Project administration, Funding acquisition; **David Costa:** Resources, Writing – review & editing, Project administration, Supervision; **André Oliveira:** Software, Conceptualization, Validation, Project administration, Funding acquisition; **Jorge Landeck:** Supervision, Project administration, Funding acquisition.

Acknowledgments

We would like to acknowledge the contributions of Orlando Sousa and Nuno Morgado, Bernardo Cabral, Nuno Peres, Artur Lopes e João António, for their support on the collection of data.

This work was supported by project SMART-PDM, n° 40123 (AAC n° 25/SI/2017) POCI-01-0247-FEDER-040123, co-funded by the European Regional Development Fund (ERDF), through the Operational Programme for Competitiveness and Internationalization (COMPETE 2020) and also by of project "FERROVIA 4.0", POCI-01-0247-FEDER- 046111, co-funded by the European Regional Development Fund (ERDF), through the Operational Programme for Competitiveness and Internationalization (COMPETE 2020), also by project OPEVA KDT JU grant nr: 101097267

References

[1] P. Chaves, et al., An IoT cloud and big data architecture for the maintenance of home appliances, in: IECON 2022 –48th Annual Conference of the IEEE Industrial Electronics Society, 2022, pp. 1–6, doi:10.1109/IECON49645.2022.9968580.
[2] T. Fonseca, L.L. Ferreira, P. Chaves, B. Cabral, P. Costa, SMART-PDM Appliance Dataset, 2022, doi:10.5281/ZENODO.7245198.
[3] T. Fonseca, L. L. Ferreira, L. Klein, J. Landeck, and P. Sousa, "Flexigy Smart-grid Architecture", doi:10.5220/0010918400003118.
[4] "SMART-PDM – A Smart Predictive Maintenance Approach based on Cyber Physical Systems." https://smart-pdm.eu/ (accessed Oct. 27, 2022).
[5] I.T. Jollife, J. Cadima, Principal component analysis: a review and recent developments, Philos. Trans. R. Soc. A 374 (2065) (Apr. 2016), doi:10.1098/RSTA.2015.0202.
[6] M. Christ, N. Braun, J. Neuffer, A.W. Kempa-Liehr, Time series FeatuRe extraction on basis of scalable hypothesis tests (tsfresh – A Python package), Neurocomputing 307 (Sep. 2018) 72–77, doi:10.1016/J.NEUCOM.2018.03.067.